Claims

1. Method for optimising compiler-generated program code (38) intended for a portable data carrier (10) having both a processor core (12) and a first and second memory area (24, 26):

- the first memory area (24) being provided to receive the optimised program code (44),

- the second memory area (26) being provided to receive a predefined library (28) having a multiplicity of library code fragments (30x), and

- the compiler-generated program code (38) being searched for program code fragments (42) which correspond, at least in respect of their effect, to respectively one library code fragment (30x), the program code fragments (42) found thereby being replaced by respectively one call of the corresponding library code fragment (30x).

2. Method according to claim 1, characterized in that a program code fragment (42) is replaced by a library code fragment (30x) only if both code fragments (42, 30x) are identical in their form as executable machine code.

3. Method according to claim 1, characterized in that at least some library code fragments (30x) are parameterised.

4. Method according to any one of claims 1 to 3, characterized in that a program code fragment (42) to be replaced is replaced, at least if the program code fragment (42) does not interfere with the program flow, by a subroutine call instruction (46) to the corresponding library code fragment (30x).

5. Method according to any one of claims 1 to 4, characterized in that the compiler-generated program code (38) exists in the form of assembler source code, and the optimisation procedure is performed on a source code level.

6.   Method according to any one of claims 1 to 5, characterized in that the predefined library (28) is matched to the hardware of the portable data carrier (10) and/or to an operating system of the portable data carrier (10) and/or to a compiler (36) used in the generation of the compiler-generated program code (38).

5

7.   Method according to any one of claims 1 to 6, characterized in that the first memory area (24) is electrically programmable, and/or the second memory area (26) is mask-programmable, and/or the first memory area (24) occupies more chip area per memory cell in the portable data carrier (10) than is occupied by the second

10   memory area (26).

8.   Computer program product comprising program instructions for a general-purpose computer which cause the general-purpose computer to execute a method according to any one of claims 1 to 7.

15

9.   Computer program product according to claim 8, characterized in that the program instructions additionally implement a compiler (36) for converting a high-level language source code (34) into the compiler-generated program code (38).

20   10.   Portable data carrier (10) having a processor core (12), a first memory area (24) and a second memory area (26), there being contained in the first memory area (24) optimised program code (44) generated by a method according to any one of claims 1 to 7, and there being contained in the second memory area (26) a library (28) which is predefined independently of the optimised program code (44) and has a

25   multiplicity of library code fragments (30x).